Assignment 3

Group members:

1) Sihan Tawsik (05)
2) Samnan Rahee (25)

Store and Bank server workflow:

The bank and the Store program communication was implemented using the JAVA Socket and JAVA ServerSocket. The servers communicate with each other using only simple text messages.

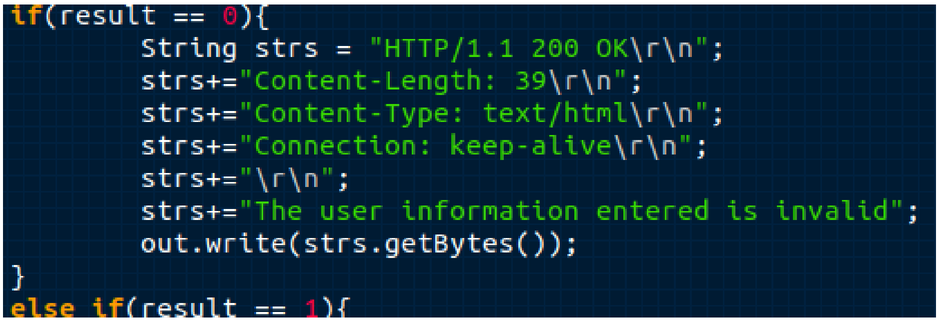
For example: `Ready, Transaction Failed, Transaction successful.`

Again, for sending user data, raw text data(space separated) was sent.

Sending response to GET command:

The response to GET command was formatted using the HTTP 1.1 format. For example:

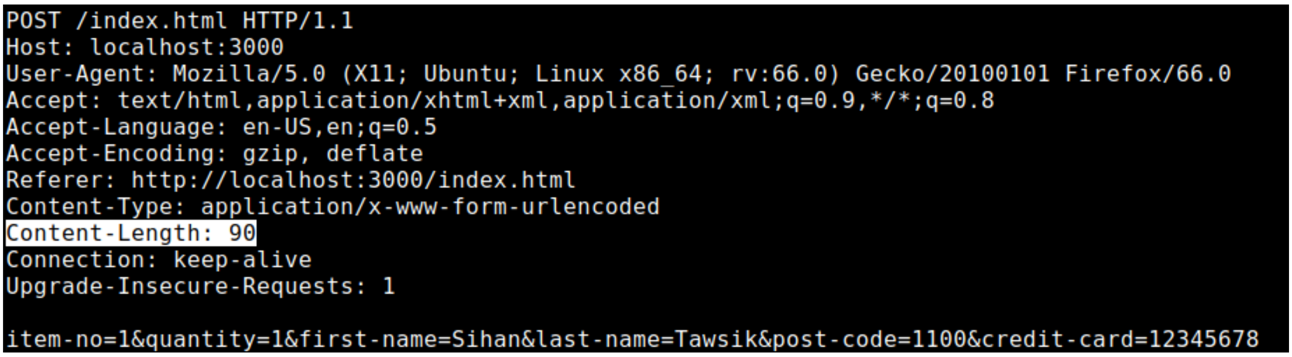
Sample response message for GET:



```
if(result == 0){
        String strs = "HTTP/1.1 200 OK\r\n";
        strs+="Content-Length: 39\r\n";
        strs+="Content-Type: text/html\r\n";
        strs+="Connection: keep-alive\r\n";
        strs+="\r\n";
        strs+="The user information entered is invalid";
        out.write(strs.getBytes());
}
else if(result == 1){
```

Handling POST requests:

Handling the POST requests were tricky. As the request body did not end with a carry return or a new line, readLine() method could not be used. Instead, it was implemented using read() method which read a single character. The submitted form data of the post request was at the bottom of the request body. And the length of the form data was given in the request body.



```
POST /index.html HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:66.0) Gecko/20100101 Firefox/66.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:3000/index.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 90
Connection: keep-alive
Upgrade-Insecure-Requests: 1

item-no=1&quantity=1&first-name=Sihan&last-name=Tawsik&post-code=1100&credit-card=12345678
```

The request body ended with newline. After that was the form data. So, same number as the content-length characters were read from the body and the data was extracted.

Usability:

The code works for every scenarios except for invalid inputs.